IJIRCCE

e-ISSN: 2320-9801 | p-ISSN: 2320-9798

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

Volume 12, Issue 5, May 2024

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

Impact Factor: 8.379

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205132 |

Language Translator Tool to Convert English to Hindi

Dr. Vijay G R, Gantolla Pavan Sai, Kushal M, Madhusudhan B, Nisarga M

Associate Professor, Department of information Science and Engineering, SJC Institute of Technology, Chickballapur, India

U.G. Student, Department of Information Science and Engineering, S J C Institute of Technology, Chickballapur, India

U.G. Student, Department of Information Science and Engineering, S J C Institute of Technology, Chickballapur, India

U.G. Student, Department of Information Science and Engineering, S J C Institute of Technology, Chickballapur, India

U.G. Student, Department of Information Science and Engineering, S J C Institute of Technology, Chickballapur, India

ABSTRACT: Machine Translation involves changing text from one language to another with the use of computer but not human participation. Machine Translation or MT is a part of Natural Language Processing and its main objective is to translate text or speech from one natural language to another using software. This process enables us overcome barriers brought about by different languages. Furthermore, through MT it is now easy to break down technological and cultural walls since people can share knowledge more often than before. The reason for this paper is try and create an automatic conversion system that turns English sentences into their Hindi equivalents. Initially we consider English-Hindi Corpus Dataset, later the words and punctuations are extracted from the file and stored in array form after which we group words together based on meaning within sentence before converting them into desired language such as Hindi. At times during this process while translating there could be word order problems; word sense disambiguation; idioms etcetera which also need attention so as to ensure correct interpretation. Finally, after obtaining translated version of given English text into Hindi, we should check if it follows grammar rules or not.

KEYWORD: Machine Translation(MT), Natural Language Processing(NLP), Text Processing, Corpus Dataset.

I. INTRODUCTION

Cross language communication plays a pivotal role in building a favorable infrastructural environment for multifaceted benefits between two countries. In this internet era, machine translation fulfills the role of an agent to perform this cross language communication. Machine translation (MT) is the use of software to translate text from one language to another. The term spans a variety of tools, with differing levels of maturity - from free, online translation tools to custom-built, industry-specific translation engines. Machine translation is the research field of Natural Language Processing (NLP) which aims to fill the gap of communication among the different of socities. After Mandrin, Spanish and English, Hindi is the most natively spoken language in the world, almost spoken by 26million people according to 2014. We chose Hindi language as target language in this paper. Recently, most of the MT works were focused on English to Indian language translation systems. However, a few systems have been constructed for English to Hindi translation, but not matured enough to resolve all inherent ambiguity and uncertainties of the Hindi sentences. As simplest level, a machine translator converts sentences by replacing word with word from source language to target language. However this is not enough because it does not take into account the semantic and syntactic restrictions of the target language. Several methods have been devised to overcome these drawbacks of automated machine translation like SMT, EBMT and RBMT. Each approach has its own pros and cons.

Hindi-English translation software are already available for free. These include Google Translator, MS-Bing and Babylon. They work on various methods such as Rule Based Machine Translation (RBMT), Example Based Machine Translation (EBMT) and Statistical Machine Translation (SMT). However, they are not very good at dealing with word sense disambiguation, idiomatic expressions translation and resolution of pronoun. Sequence to sequence neural networks are making that dream a reality! These clever models are a powerful type of deep learning architecture that can transform sequences of data, like sentences in English, into new sequences, like their translations in Hindi. Imagine a machine that a reads English sentence, understands its meaning, and then rewrites it entirely in Hindi. That's the magic of seq2seq models at work. This introduction will unveil the inner workings of seq2seq, exploring how they

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205132 |

break down information, translate it, and generate completely new sequences, opening a world of possibilities for machine translation, text summarization, and other exciting applications.

II. LITERATURE REVIEW

This survey explains the information of machine translation within the lapse of time. The research has been done in machine translation and found that there are existing machine translation systems in various regions of the world. Machine Translation began after second world war during 1950's. There was research done from the Georgetown University, where they worked with the translation system which translates the sentences of the Russian language into the English. This survey outlines the information about the different approaches to machine translation, as well as related works.

A. Rule-Based Machine Translation

Jordi et al. have used Rule Based Machine Translation for Chinese to Spanish Machine translation. In this work, they have used Apertium platform which is a toolbox for shallow transfer MT. For the generation of translation rules, a bilingual Chinese-Spanish dictionary is constructed consisting of almost 9000 distinctive words. Grammatical transfer-rules were developed manually. They test this system on different domains with average accuracy of 82%. First, the system analyzes the source language sentence, using dictionaries to find corresponding words in the target language. Then, grammar rules for both languages come into play. These rules tell the system how to rearrange and adjust the words to form a proper sentence in the target language, ensuring it conveys the same meaning as the original sentence. Finally, the translated sentence is generated, following the target language's grammatical structure.

B. Corpus-Based Machine Translation

Corpus-Based Machine Translation As part of machine translation, corpus-based approaches have emerged as an area that has been extensively explored since 1989. This method has been favoured over others because of its high level of accuracy. In contrast to RBMT, corpus-based machine translation eliminates the Knowledge acquisition problem. To achieve the translation, the system uses a large bilingual parallel corpus. Translations are made using a system that translates source language to target language. This intermediate form will then generate an output text. By recognizing these patterns, the model can then predict how to translate new, unseen text in the source language. There are different approaches within corpus-based translation, but they all leverage this core idea of learning from a large dataset of existing translations. The two main types of CBMT systems are Statistical Machine Translation (SMT) and Example Based Machine Translation (EBMT).

C. Statistical Machine Translation

Statistical Machine Translation (SMT) is dependent on statistical methods Philipp et al (2007), Richard et al (2002), Mary et al (2011). It is a data driven technique that makes use of parallel-aligned corpora. A statistical approach based English-to-Hindi machine translation system is developed consisting of three processing units Language Model, Translation Model and Decoder. Language model calculates the probability of a sentence in target language. First, SMT systems go through a massive amount of text that's already been translated by humans, like documents or articles. This bilingual data helps the system identify patterns between the source language and the target language. By crunching these numbers, SMT builds a model that predicts the most likely translation for a new piece of text. The model considers both the translation itself (how probable a certain phrase is in the target language) and how well it fits the context of the surrounding words. This way, SMT can translate text while also trying to preserve the original meaning.

D. Example Based Machine Translation

Example based machine translation systems (EBMT) perform the translation of a given input sentence s in three consecutive phases (i) (matching) check for existence of the given input s in the bilingual corpus (ii) (retrieval) extraction of useful segments from the sentence that match in the bilingual corpus and (iii) (transfer) recombining the translated segments. Example-Based Machine Translation (EBMT) works by learning from pre-translated examples. Imagine a translator having a giant library of phrases and sentences they've translated before, along with their corresponding translations in another language. When given a new sentence to translate, the EBMT system searches this library for similar examples. It then analyzes these matches and uses them to assemble the most fitting translation for the new sentence. By reusing past translations as a guide, EBMT aims to deliver accurate and natural-sounding translations, especially for common phrases and sentence structures.

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205132 |

E. Hybrid Machine Translation (HMT).

Hybrid machine translation is a method of machine translation where the characteristics of different machine translation approaches are integrated into a single machine translation system. There are two main ways this can work. In one approach, the system might run different translation engines, like rule-based and statistical ones, at the same time. Then, it would combine the best parts of each translation to create the final output. Another approach involves using one type of engine first, like a rule-based engine, and then using another type, like a statistical engine, to improve the accuracy and fluency of the translation. Paul et al. also came up with a multi-engine hybrid approach to Machine Translation, which uses statistical models to derive the best output from multiple machine translation systems. He has obtained very promising results on Japanese-English machine translation using a decision-tree method for selecting the best theoretically possible hypothesis attained from multiple RBMT, EBMT and SMT decoders.

III. CHALLENGES WITH EXISITING SYSTEMS

This section discusses some challenges that in English to Hindi translation systems poses the challenges of structural and morphological differences. In the following subsections, we discuss the syntactic and morphological divergences with examples.

[i] Morphological Differences: Here we discuss about morphological Divergence between English with Hindi analyzing with large parallel corpus data which has a more complex system of word forms and word endings compared to English This means that translating between the two languages requires Careful handling of these morphological variations.

[ii]Structural Differences: Here we discuss about the important Structural Differences the basic word order in English is Subject-Verb-Object (SVO), whereas in Hindi it is Subject-Object-Verb (SOV). This difference in sentence structure can pose difficulties when trying to accurately convey the meaning of a sentence from one language to the other.

These inherent divergences between English and Hindi are the primary reasons why machine translation systems face challenges in accurately translating between the two languages.To elaborate more, when a sentence in one language L1 is translated into another language L2, it can be seen in a very different way . For a strong machine translation (MT) system, it's important not only to recognize the kinds of translation differences but also to fix them to get better translations. In this paper, several English-Hindi translation divergences have been studied in order to identify language specific divergences and further to be incorporated in EBMT and RBMT phases during the translation. In terms of configurational characteristics, In below example "कल" is used to address both tomorrow and as well as yesterday.

Example 1. a. "Tomorrow" ⇒ "कल"(kal)
b. "Yesterday" ⇒ "कल"(kal)

The conversation above shows how important it is to look at all kinds of differences in translation. Doing this will make a strong English-Hindi MT. It's hard to handle all these differences at once. As, it is difficult to process and deal with the all type of translation divergences simultaneously, we have tried to incorporate neural network machine translation.

IV. PROPOSED APPROACH

This article throws light on various aspects of building a basic Neural Machine Translation (NMT) model using the Sequence to Sequence learning approach, with LSTM. While there are numerous papers and blogs on NMT, this is another attempt on highlighting some of the intuitive features and also a step by step guide to performing similar NLP tasks

A. The Corpus-Dataset:

The experiments in this paper use the IIT Bombay English-Hindi Parallel Corpus. This dataset consists of an English-Hindi Parallel Corpus that has been pre-processed for machine translation. This is a Hindi-English parallel corpus containing **1,492,827** pairs of sentences. The Hindi side of the training, dev, test sets as well as the monolingual corpus have been normalized to ensure canonical Unicode representation using the Indic NLP Library. The moses tokenizer for English, and the Indic NLP tokenizer for Hindi was used to prepare the data. In addition to the challenge of handling

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205132 |

two linguistically and semantically different l languages, there were other pre-processing tasks required like: The data needs some minimal cleaning before being used to train a neural translation model. [i] Tokenizing text by white space. [ii] Normalizing case to lowercase. [iii] Removing punctuation from each word. [iv] Removing non-printable characters. [v] Removing words that contain non-alphabetic characters.

B. Preparing Training Data:

We still have the data in text format. We need to make it machine-ready for training our model. So, before model design, we will perform Tokenizing and Indexing (you can use NLTK tokenizers or others available from Indic NLP library). For tokenization, we will find all the unique words in both languages. The arrays, one for encoder input, one for decoder input and one for decoder target. Which will be used this for indexing each word. The dimensions 30 and 32 are because of the maximum sentence lengths we have decided. It is 30 for encoder (English) and 32 for decoder (Hindi). Decoder limit is 32 because "START_" and "_END" are appended to the beginning and the end of the target sentences (in this case, Hindi), so that the decoder has a stopping condition which is either an "_END" being encountered, or maximum word limit is reached. Also, the "START_" is used because decoder output will be one time-step ahead. This step sounds complicated, but all we are doing is, breaking the sentences and assigning an integer to each unique word, mostly creating a dictionary. The following example can help explain this:

```
Sentence:       This is me
Tokenization:   ['This','is','me']
 Indexing:        This -> 1
                  is -> 2
                  me -> 3
```

C. Sequence to Sequence model with Attention Mechanism:

A significant hurdle in sequence-to-sequence learning is that the model typically requires both the input and output sequences to have a fixed and predetermined length. To address this limitation, researchers implemented Long Short-Term Memory (LSTM) networks within the sequence-to-sequence architecture. LSTMs are adept at capturing the meaning of a sequence and mapping sentences with similar meanings close together. This allows the sequence-to-sequence model with LSTM to account for word order and effectively handle both active and passive voice constructions. The architecture involves an encoder and a decoder. The encoder consists of the following layers:

[i] Embedding Layer: The embedding layer plays a crucial role in machine translation models by transforming individual words from the source language (English in this case) into numerical vectors with a fixed size. These vectors aim to capture the meaning and semantic relationships between words. There's flexibility in how you use the embedding layer. You can train it independently to learn these word embeddings beforehand, and then use those pre-trained vectors in your model. Alternatively, the embedding layer can be part of the overall model, where the word vectors are learned as the model itself is trained. Pre-trained vectors from algorithms like Word2Vec or GloVe can also be used within this layer, leveraging the existing knowledge these algorithms have captured about word relationships.

[ii] LSTM Layer: Recurrent Neural Networks (RNNs) like LSTMs (Long Short-Term Memory networks) process information sequentially. Unlike feedforward networks that process all inputs at once, LSTMs handle data one element (word) at a time. This allows them to capture the relationships between words in a sequence, which is crucial for tasks like machine translation or sentiment analysis. With each word processed, the LSTM layer updates its internal state, essentially remembering what it has seen so far. This internal state is then combined with the new word to create a fixed-size vector representation that reflects the meaning of the sequence seen up to that point. LSTMs can have multiple layers stacked on top of each other, allowing them to learn increasingly complex relationships within the data. This stacking helps capture long-term dependencies within sequences, even for very long sentences.

Just like Encoder corresponding Decoder part of LSTM works as follows: The decoder plays a crucial role in generating the target language sentence. Similar to the encoder, it utilizes a stack of LSTM units. However, unlike the encoder that processes the entire input sentence at once, the decoder operates in a step-by-step manner. It leverages the context vector, which captures the essence of the source language, to predict the target sentence word by word. At each step, the decoder's LSTM unit considers both the context vector and the previously generated words to determine the most likely next word in the target language. This dependency on past outputs makes LSTMs well-Suited for this task, as they can bridge the gap between the source sentence and the unfolding target sentence despite the potential delay between them.

International Journal of Innovative Research in Computer and Communication Engineering

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205132 |

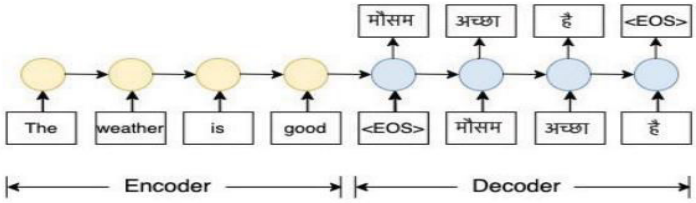

Fig. 1. Sequence to Sequence architecture using LSTM

In Fig. 1, the input sentence in the English language fed to the encoder "The weather is good" is translated to Hindi sentence "मौसम अच्छा है" and generated by the decoder.

The decoder uses the context vector, the previous words are being given equal importance for the prediction of the next word. But the meaning of the next word may depend on some specific words instead of all previous words. A regular LSTM Seq2Seq model struggles with long sentences because it crams all the information into a single context vector. This vector might not capture everything important. To address this, attention mechanisms were introduced. Attention allows the model to focus on specific parts of the input sentence at each step while decoding the output. Instead of relying solely on the final context vector, the attention mechanism creates a score for each part of the input sentence. These scores indicate how relevant each part is for predicting the current word in the output sequence. The model then uses these scores to create a weighted context vector, focusing on the informative parts of the input sentence. In simpler terms, attention helps the model pick out the most important details from long sentences, one word at a time, resulting in more accurate translations or outputs.

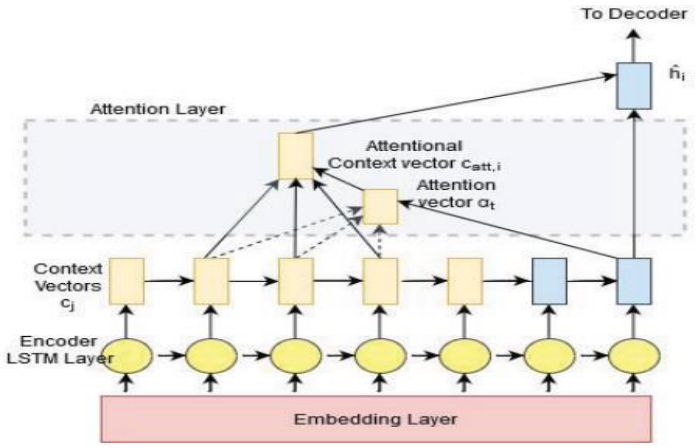

Fig. 2. Attention Model

Fig. 2 shows that an attentional context vector is generated using the previous context vectors generated by the decoder and the attention vector. The attention weights compute the importance of the source word corresponding to the target word generation using alignment model as shown in Equation (1).

$$e_{i,j} = align(z_{i-1}, h_j), \quad \forall j \in 1, 2, ...T, \quad \forall i \in 1, 2, ...T' \qquad (1)$$

Where $e_{i,j}$ is the alignment score of j-th source word corresponding to i-th target word, Z_{i-1} is decoders last state, h_j. The annotation vector of j-th source word and T and T^1 are the lengths of source and target sentences. Afterward, alignment scores are converted into probabilistic measures, are called attention weights ($\alpha_{i,j}$) as given in the Equation (2).

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205132 |

$$\alpha_{i,j} = \frac{exp(e_{i,j})}{\sum_{j'} exp(e_{j',i})} \quad (2)$$

Finally, the context vector is computed using the annotation vector (ci) as shown in Equation (3).

$$c_i = \sum_{j=1}^{T} \alpha_{i,j} \times h_j \quad (3)$$

After context vector is computed then calculate decoders next state as a non-linear function (in case of LSTM) ocontext vector c_i , previous target word μ_{i-1} and decoders last state z_{i-1} using Equation (4).

$$z_i = f(c_i, \mu_{i-1}, z_{i-1}) \quad (4)$$

D. System Testing: System training is followed by the system testing/translation process where beam search, an optimized heuristic best first search technique is used to search the finest translations.

V. RESULTS AND DISCUSSION

A. Data Description: A English-Hindi bilingual dictionary by CFILT, IIT-Bombay is used to translate words between Hindi and English. This dictionary has over 136,000 English words with their Hindi meanings. Another resource used for translation is a collection of parallel sentences, English on one side and Hindi on the other. This collection includes nearly 290,000 sentences and helps with machine translation techniques.

Table I:

HINDIENCORP CORPORA STATISTICS

Language Units	English	Hindi
Token	2,898,810	3,092,555
Types	95,551	118,285
Total Characters	18,513,761	17,961,357
Total Sentences	289,832	289,832
Sentences (word count $\leq$ 10)	188,993	182,777
Sentences (word count $>$ 10)	100,839	107,055

B. Results

Neural machine translation (NMT) with sequence-to-sequence (seq2seq) and attention mechanisms has revolutionized machine translation. The attention mechanism allows the decoder to focus on specific parts of the source sentence while generating the target sentence. This leads to more accurate and nuanced translations, especially for complex or lengthy sentences.

English	Hindi
India is my country	भारत मेरा देश ह
Do you see the man who is running across the road?	क्या आप उस आदमी को देखते हैं जो सड़क पर चल रहा है?
The region shares its borders with China Nepal, Bhutan	यह क्षेत्र चीन, नेपाल, भूटान के साथ अपनी सीमाओं को साझा करता है
It is not at the centre of the paper	यह कागज़ के केंद्र में नहीं है।

Table II: Few Examples for English-Hindi Machine Translation

| e-ISSN: 2320-9801, p-ISSN: 2320-9798| www.ijircce.com | |Impact Factor: 8.379 | Monthly Peer Reviewed & Referred Journal |

|| Volume 12, Issue 5, May 2024 ||

| DOI: 10.15680/IJIRCCE.2024.1205132 |

VI. CONCLUSION

The results of a neural machine translation (NMT) system that uses a sequence-to-sequence recurrent neural network (RNN) model. The system was used to translate text from English to Hindi, and the when compared the output of their system to existing machine translation (MT) systems using a metric called BLEU score. Their system showed better performance on this metric than existing systems. However, there is room for improvement in their NMT system. In particular, they found that the system struggled to recognize unknown words and to generate diverse translations of the source sentence. These are both common challenges in machine translation, and there is ongoing research into how to improve NMT systems in these areas. Overall, the results of this study suggest that NMT systems with seq2seq and attention mechanisms are a promising approach to machine translation. However, there is still work to be done in order to improve the accuracy and fluency of these systems.

REFERENCES

[1] Laskar, Sahinur Rahman, et al. "Neural machine translation: English to hindi." 2019 IEEE conference on information and communication technology. IEEE, 2019.
[2] Dhariya, Omkar, Shrikant Malviya, and Uma Shanker Tiwary. "A hybrid approach for Hindi-English machine translation." 2017 international conference on information networking (ICOIN). IEEE, 2017.
[3] Tan, Zhixing, et al. "Neural machine translation: A review of methods, resources, and tools." AI Open 1 (2020): 5-21.
[4] Singh, Aryan, and Jhalak Bansal. "Neural machine transliteration of indian languages." 2021 4th International Conference on Computing and Communications Technologies (ICCCT). IEEE, 2021.
[5] Tiwari, Gaurav, et al. "English-Hindi neural machine translation-LSTM seq2seq and ConvS2S." 2020 International Conference on Communication and Signal Processing (ICCSP). IEEE, 2020.
[5] Sindhu, C., Soumyajit Guha, and Yuvraj Singh Panwar. "English to Hindi translator using Seq2seq model." 2022 8th International Conference on Advanced Computing and Communication Systems (ICACCS). Vol. 1. IEEE, 2022.
[6] Patel, Raj Nath, Prakash B. Pimpale, and M. Sasikumar. "Machine translation in Indian languages: challenges and resolution." Journal of Intelligent Systems 28.3 (2019): 437-445.
[7] Singhal, Aakrit. "Effective Approaches and Challenges in Hindi-English Neural Machine Translation." (2021).
[8] Shalu, P., and M. Meera. "Neural machine translation for english to hindi using gru." Available at SSRN 3851323 (2021).
[9] Neubig, Graham. "Neural machine translation and sequence-to-sequence models: A tutorial." arXiv preprint arXiv:1703.01619 (2017).
[10] Hettige, B., and A. S. Karunananda. "Existing systems and approaches for machine translation: A review." Srilanka Association for Artificial Intelligence, Eight Annual Session (2011).
[11] Mall, Shachi, and Umesh Chandra Jaiswal. "Survey: machine translation for Indian language." International Journal of Applied Engineering Research 13.1 (2018): 202-209.
[12] Zhang, Biao, Deyi Xiong, and Jinsong Su. "Neural machine translation with deep attention." IEEE transactions on pattern analysis and machine intelligence 42.1 (2018): 154-163.
[13] Liu, Lemao, et al. "Neural machine translation with supervised attention." arXiv preprint arXiv:1609.04186 (2016).
[14] He, Weihua, Yongyun Wu, and Xiaohua Li. "Attention mechanism for neural machine translation: a survey." 2021 IEEE 5th Information Technology, Networking, Electronic and Automation Control Conference (ITNEC). Vol. 5. IEEE, 2021.
[15] Shachi Dave, Jignashu Parikh, and Pushpak Bhattacharyya, "Interlingualbased English-hindi machine translation and language divergence," Machine Translation, vol. 16(4), pp.251–304 (2001).
[16] Kyunghyun Cho, Bart van Merrienboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio, "Learning phrase representations using rnn encoderdecoder for statistical machine translation," In Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP), Association for Computational Linguistics, Doha, Qatar, pp. 1724–1734 (2014).
[17] lya Sutskever, Oriol Vinyals, and Quoc V. Le, "Sequence to sequence learning with neural networks," In proceedings of the 27th international conference on neural information processing systems - Volume 2. MIT Press, Cambridge, MA, USA, NIPS14, pp. 3104–3112 (2014).
[18] Holger Schwenk, "Continuous space translation models for phrasebased statistical machine translation," In Proceedings of COLING 2012: Posters. The Coling 2012 Organizing Committee, Mumbai, India, pp. 1071–1080 (2012)

INTERNATIONAL STANDARD SERIAL NUMBER INDIA

INNO SPACE SJIF Scientific Journal Impact Factor

doi crossref

निस्केयर NISCAIR

INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH

IN COMPUTER & COMMUNICATION ENGINEERING

9940 572 462 6381 907 438 ijircce@gmail.com

www.ijircce.com

Scan to save the contact details